

## PATENT ABSTRACTS OF JAPAN

(11)Publication number : 2004-178015

(43)Date of publication of application : 24.06.2004

(51)Int.Cl.

G06F 12/00

(21)Application number : 2002-340041

(71)Applicant : TOSHIBA CORP

(22)Date of filing : 22.11.2002

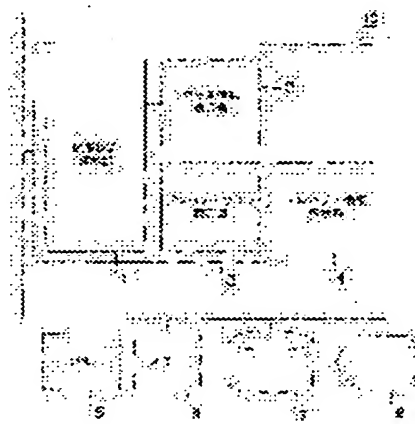
(72)Inventor : MINAMINO NORIKO  
MIZOGUCHI YUMIKO  
MURAYAMA HIROSHI

## (54) HIERARCHICAL STRUCTURE DISPLAY DEVICE AND HIERARCHICAL STRUCTURE DISPLAY METHOD

## (57)Abstract:

**PROBLEM TO BE SOLVED:** To provide a hierarchical structure display device which adequately and effectively displays inclusive relationships among classes in a hierarchical database, and also provide a method therefor.

**SOLUTION:** When the hierarchical structure of a hierarchical database in which classes having a hierarchical structure are defined is displayed, first and second display areas are displayed so that at least a part of the first display area identifying one of the classes having a hierarchical structure includes the whole of the second display area identifying a slave class derived from the above class.



(19) 日本国特許庁 (JP)

## (12) 公開特許公報(A)

(11) 特許出願公開番号

特開2004-178015

(P2004-178015A)

(43) 公開日 平成16年6月24日 (2004. 6. 24)

(51) Int. Cl.<sup>7</sup>

G06F 12/00

F I

G06F 12/00

5 4 7 A

G06F 12/00

5 1 5 B

テーマコード (参考)

5 B 0 8 2

審査請求 未請求 請求項の数 15 O L (全 16 頁)

(21) 出願番号 特願2002-340041 (P2002-340041)  
(22) 出願日 平成14年11月22日 (2002. 11. 22)(71) 出願人 000003078  
株式会社東芝  
東京都港区芝浦一丁目1番1号  
(74) 代理人 100058479  
弁理士 鈴江 武彦  
(74) 代理人 100084618  
弁理士 村松 貞男  
(74) 代理人 100068814  
弁理士 坪井 淳  
(74) 代理人 100092196  
弁理士 橋本 良郎  
(74) 代理人 100091351  
弁理士 河野 哲  
(74) 代理人 100088683  
弁理士 中村 誠

最終頁に続く

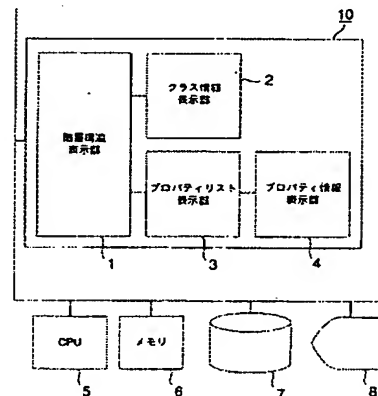
(54) 【発明の名称】 階層構造表示装置および階層構造表示方法

## (57) 【要約】

【課題】 階層型データベースにおけるクラス間の包含関係を適切かつ効果的に表示することのできる階層構造表示装置および方法を提供すること

【解決手段】 階層構造をなすクラスが定義された階層型データベースの階層構造表示を表示するにあたり、階層構造をなすいずれかのクラスを識別する第1の表示領域の少なくとも一部が、該クラスから派生した子クラスを識別する第2の表示領域の全てを含むように前記第1及び第2の表示領域を表示する。

【選択図】 図1



## 【特許請求の範囲】

## 【請求項 1】

階層構造をなすクラスが定義された階層型データベースの階層構造表示装置において、前記階層構造をなすいずれかのクラスを識別する第 1 の表示領域の少なくとも一部が、該クラスから派生した subclasses を識別する第 2 の表示領域の全てを含むように前記第 1 及び第 2 の表示領域を表示する階層構造表示手段を具備することを特徴とする階層構造表示装置。

## 【請求項 2】

選択された前記一のクラス及び subclasses のいずれかに関連するクラス情報を表示するクラス情報表示手段を具備することを特徴とする請求項 1 に記載の階層構造表示装置。

10

## 【請求項 3】

選択された前記一のクラス及び subclasses のいずれかが有するプロパティのリストを前記クラス情報と関連付けて表示するプロパティリスト表示手段を具備することを特徴とする請求項 2 に記載の階層構造表示装置。

## 【請求項 4】

前記プロパティリストから選択されたプロパティに関連するプロパティ情報を表示するプロパティ情報表示手段を具備することを特徴とする請求項 3 に記載の階層構造表示装置。

## 【請求項 5】

継承元のクラスと、継承先のクラスにおける前記プロパティリスト中のプロパティとを同一色で対応付けて表示する手段を具備することを特徴とする請求項 3 に記載の階層構造表示装置。

20

## 【請求項 6】

前記 subclasses を識別する第 2 の表示領域の表示有無を前記一のクラスに対して与えられた開閉操作に応じて切り替える手段を具備することを特徴とする請求項 1 乃至 5 のいずれかに記載の階層構造表示装置。

## 【請求項 7】

インスタンスを有するクラスのみを展開表示する手段を具備することを特徴とする請求項 1 乃至 6 のいずれかに記載の階層構造表示装置。

## 【請求項 8】

前記階層構造内において選択された一のクラスの直系クラスのみを表示する手段を具備することを特徴とする請求項 1 乃至 7 のいずれかに記載の階層構造表示装置。

30

## 【請求項 9】

前記いずれかのクラスに関連付けて、前記 subclasses のいずれか一つがインスタンスを有するか否かを識別するマークを表示する手段を具備することを特徴とする請求項 1 乃至 8 のいずれかに記載の階層構造表示装置。

## 【請求項 10】

前記階層構造内のいずれかのクラスを検索開始ポイントに設定する手段と、前記検索ポイントを設定するクラスをインスタンスを有するクラスに限定する手段とを具備することを特徴とする請求項 1 乃至 9 のいずれかに記載の階層構造表示装置。

## 【請求項 11】

前記検索ポイントを設定するクラスを、インスタンスを有する少なくとも 2 つ以上の子クラスを有するクラスに限定することを特徴とする請求項 10 に記載の階層構造表示装置。

40

## 【請求項 12】

前記階層型データベースは、複数の親クラスからのプロパティの多重継承を許容することを特徴とする請求項 1 乃至 11 のいずれかに記載の階層構造表示装置。

## 【請求項 13】

前記多重継承に係るプロパティを選択し、該プロパティの継承元に相当する前記親クラスのいずれかを参照する手段を具備することを特徴とする請求項 12 に記載の階層構造表示装置。

## 【請求項 14】

50

第1のルートを有する第1の階層構造と、第2のルートを有する第2の階層構造とに共通するユニバーサル・ルートを設定する手段を具備することを特徴とする請求項1乃至13のいずれかに記載の階層構造表示装置。

【請求項15】

階層構造をなすクラスが定義された階層型データベースの階層構造表示方法において、前記階層構造をなすいずれかのクラスを識別する第1の表示領域の少なくとも一部が、該クラスから派生した子クラスを識別する第2の表示領域の全てを含むように前記第1及び第2の表示領域を表示する階層構造表示ステップと、  
前記一のクラス及び子クラスのいずれかの選択を受け付けるステップと、  
前記選択に連動し、該選択された前記一のクラス及び子クラスのいずれかに関連するクラス情報を表示するクラス情報表示ステップと、  
前記選択に連動し、該選択された前記一のクラス及び子クラスのいずれかが有するプロパティのリストを前記クラス情報と関連付けて表示するプロパティリスト表示ステップとを具備することを特徴とする階層構造表示方法。

10

【発明の詳細な説明】

【0001】

【発明の属する技術分野】

本発明は、階層型データベースに好適な階層構造表示装置および方法に関する。

【0002】

【従来の技術】

今日、コンピュータソフトウェアの技術分野は、よりユーザフレンドリな機能を提供すべく日々進歩を遂げている。例えば、ファイルシステムの利用にあたりユーザが所望のディレクトリに移動して所望のファイルを開くような操作（オペレーション）を視覚的かつ直感的に行えるように、グラフィック・ユーザ・インタフェース（GUI）は線やアイコン等を組み合わせてディレクトリ（フォルダ）構造をツリー状に表示し、ユーザに提示することができる。このようなGUI機能はマイクロソフト社のオペレーティング・システム（OS）Windows（商標）や、その他にもUNIX（商標）やLINUX（商標）などの汎用のオペレーティングシステムにおいて広く採用されている。

20

【0003】

ディレクトリ構造ツリーの各ノード（ディレクトリ）間について、上位のノードに含まれる情報（ファイル等）と下位のノードに含まれる情報との間に、包含あるいは部分集合といった関係はない。すなわち、ディレクトリ構造ツリーを形成する、ルート・ノードから始まる各ノードは、ファイルなどの情報を納めるコンテナの相互接続関係を表しているにすぎない。

30

【0004】

一方、オブジェクト指向データベース（OODB）やオブジェクト・リレーショナルデータベース（ORDB）を代表とするデータベースは、階層構造を有し、該階層構造において下位分類が上位分類の属性を継承する仕組みを有する。このようなデータベースでは、継承によって下位の分類では属性が累増するという特徴がある。下位分類が上位分類の属性を継承することは、「インヘリタンス」とも呼ばれ、このような技術は多くの文献に記載されている（例えば下記非特許文献1参照。）。

40

【0005】

なお、オブジェクト指向データベース（OODB）関連の技術分野では、階層中の分類を「クラス」と呼ぶことが多い。本明細書では、「分類」と「クラス」とをほぼ同じ意味を有する用語として用いる。また、オブジェクト・リレーショナル型データベース（ORDB）においては、継承を許したテーブルがクラスに相当する。上下関係にあるテーブル間においては、上位のテーブルから下位のテーブルへ属性が継承される。ここでの属性は、ORDBにおいては上位テーブルを構成するコラムのヘッダー情報に相当し、これが下位テーブルへ継承される。

【0006】

50

本明細書においては、オブジェクト指向データベース（OODB）とオブジェクト・リレーショナルデータベース（ORDB）の両者を含めて「階層型データベース」と称する。また、各階層のクラスに属する同じ属性種を持つデータを「インスタンス」と呼び、その集合をデータの「ポピュレーション」と呼ぶ。

【0007】

このように、階層型データベースの階層構造は、クラス間の包含関係（継承関係）があることから、上述のディレクトリ構造ツリーとはその性質が明らかに異なるものである。

【0008】

階層型データベースを製品カタログデータベースに応用したサービスが既に幾つか提供されている（下記非特許文献2、3を参照。）。

【0009】

【非特許文献1】

Object-Oriented Concepts, Databases, and Applications, Edited by Won Kim, 1989, ACM Press

【0010】

【非特許文献2】

株式会社ポータル、インターネット<URL: http://www.portalcorp.net>

【0011】

【非特許文献3】

e-ingBiz.com、インターネット<URL: http://www.e-ingbiz.com/CATALOG2/servlet/CatalogSearch?Lang=ja>

【0012】

【発明が解決しようとする課題】

階層構造表示に関し、従来のディレクトリ構造ツリーにおける表示、例えばマイクロソフト社（商標）のエクスプローラでは、

- ・下位のフォルダ（ディレクトリ）を上位のフォルダへの開閉要求に応じて展開表示する
- ・段下げにより、そのフォルダよりが上位のフォルダに対して下位のフォルダであることを表す

- ・階層のレベル（深さ）ごとに段下げを揃えることにより、レベルが同じフォルダであることが容易に分かるようにする

といったことが行える。

【0013】

しかしながら、これらはあくまでディレクトリ（ノード）の接続関係を辿ることができるようにするものでしかない。

【0014】

したがって、このような従来の階層構造表示は、上位のクラスから下位のクラスへプロパティ（属性）が継承されるような階層型データベースには適さない。

【0015】

階層型データベースにおける階層構造表示には、個々のクラスが、その所有するプロパティによって特徴付けられる概念を表す場合の、クラス間の包含関係を表現できることが求められる。また、階層型データベースの階層構造がクラスによって構成される場合、該クラスのインスタンスの存在有無を表示できることが好ましい。

【0016】

本発明はかかる事情を考慮してなされたものであり、階層型データベースにおけるクラス間の包含関係を適切かつ効果的に表示することのできる階層構造表示装置および方法を提供することを目的とする。

【0017】

10

20

30

40

50

## 【課題を解決するための手段】

本発明に係る階層構造表示装置は、階層構造をなすクラスが定義された階層型データベースの階層構造表示装置において、前記階層構造をなすいずれかのクラスを識別する第1の表示領域の少なくとも一部が、該クラスから派生した子クラスを識別する第2の表示領域の全てを含むように前記第1及び第2の表示領域を表示する階層構造表示手段を具備することを特徴とする。

## 【0018】

本発明に係る階層構造表示方法は、階層構造をなすクラスが定義された階層型データベースの階層構造表示方法において、前記階層構造をなすいずれかのクラスを識別する第1の表示領域の少なくとも一部が、該クラスから派生した子クラスを識別する第2の表示領域の全てを含むように前記第1及び第2の表示領域を表示する階層構造表示ステップと、前記一のクラス及び子クラスのいずれかの選択を受け付けるステップと、前記選択に連動し、該選択された前記一のクラス及び子クラスのいずれかに関連するクラス情報を表示するクラス情報表示ステップと、前記選択に連動し、該選択された前記一のクラス及び子クラスのいずれかが有するプロパティのリストを前記クラス情報と関連付けて表示するプロパティリスト表示ステップとを具備することを特徴とする。

## 【0019】

## 【発明の実施の形態】

以下、図面を参照しながら本発明の実施形態を説明する。

## 【0020】

図1は本発明の一実施形態に係る階層構造表示システムの概略構成を示すブロック図である。本システムは、階層型データベースの階層構造表示に基づいて、ユーザが検索やデータベース管理等を、その規模や詳細度等に依らず適切かつ容易に行えるようにするものである。本発明に係る階層構造表示装置は、CPU5、メモリ6、二次記憶装置7、ディスプレイ8等の基本ハードウェアを備えた汎用のコンピュータを用い、同コンピュータ上で動作するアプリケーションプログラム10として実現することができる。この場合、同コンピュータにはアプリケーションプログラム10の実行環境を提供するとともに、ハードウェアを制御するためのオペレーティングシステムも導入されている。階層構造表示装置のアプリケーションプログラム10は、オペレーティングシステム上に構築され、ディスプレイ8や図示しないマウス、キーボード等と協働するグラフィカルユーザインターフェース（GUI）を構成する。階層型データベースは、例えば二次記憶装置7に導入されている。

## 【0021】

同図に示すように階層表示装置は、階層構造表示部1と、クラス情報表示部2と、プロパティリスト表示部3と、プロパティ情報表示部4とにより構成されている。

## 【0022】

ISO13584に規定のように、プロパティなどのクラス特性を下位クラスが上位クラスから継承するような階層型データベースにおいては、階層中の単なる上位、下位の関係ではなく、包含関係として階層構造が表現されたほうが、利用者にとってはクラス間の関係がわかりやすい。

## 【0023】

そこで、階層構造表示部1は、図2に示すようにクラス間の包含関係を明示的に表示する。具体的には、上位クラスが下位クラスを包含するように矩形等のダイアグラムを用いてクラスを表示する。

## 【0024】

図2の表示例によれば、"Product"クラス20が"Vehicle"クラス21を包含していることが明示される。この"Vehicle"クラス21は、"AutoBicycle"クラス22および"AutoMobile"クラス23の2つのクラスを含んでいることがわかる。言い替えれば、"AutoBicycle"クラス22および"AutoMobile"クラス23の2つのクラスは、"Vehicle"クラス21

10

20

30

40

50

に包含されている。さらに、"AutoMobile" クラス 23 は "Passenger Car" クラス 24 および "Truck" クラス 25 を包含していることもわかる。

【0025】

このような矩形ダイアグラムを用いた包含関係の表示は、次のように変形することができる。例えば、図 2 のような包含関係を表すダイアグラムの一部、例えば紙面右側の端部や、紙面下側の端部を切り欠く（省略）表示としてもよい。ユーザはこれらの切り欠き部分を容易に推測できることから、必ずしも包含関係が不明瞭とはならない。このような表示によれば表示ウィンドウ上の占有領域を節約できる。なお、切り欠き表示の領域はマウス等によるウィンドウバーへの操作にダイナミックに連動して調整できることが好ましい。

【0026】

階層型データベースの各クラスは、インスタンスを持つことができる。インスタンスは実例ともいう。階層構造表示部 1 は、当該クラスがインスタンスを持つか否かを上記包含関係とともに明示的に表示する。

【0027】

従来の階層構造表示では、ユーザに対してクラスの開閉の指針が示されているような表示例はない。つまり、クラスを上位から下位に向かって開いてみなければ、下位のクラスにおけるインスタンスの存在有無を確認できない。

【0028】

本実施形態では、図 2 中の傘マークのようなマークを付けることにより、クラスの開閉を行わなくとも、上位のクラスでインスタンスの存在が確認できる。また、カバンマークのようなマークを付けることにより、明確にインスタンスの存在が確認できる。このようなインスタンスの有無表示によれば、ユーザは絞り込み検索を行いやすくなる。

【0029】

図 2 の例において、傘マーク（例えば参照数字 30）は、下位に（子）クラスが存在することを表し、鞆マーク（例えば参照数字 32, 33）は、クラスにインスタンスが存在することを表している。傘マークが付与されていないクラス（例えば "Sedan" クラス 29 や "Truck" クラス 25）は、末端クラスであり、すなわち下位クラスを持たない。

【0030】

閉じた傘マーク（例えば参照数字 31）は、その下位のクラス階層を開いていない状態を表す。この閉じた傘マーク 31 や 34 は、傘の色が無色（或いは白抜き）となっており、その下位クラスにインスタンスが存在しないことを明示している。例えば、"AutoBicycle" クラス 22 以下には下位クラスが存在するが、傘マーク 31 が閉じているため表示されていないことがわかる。また、この "AutoBicycle" クラス 22 には、登録されているインスタンスはないことがわかる。

【0031】

閉じた傘マークで有色のもの（例えば参照数字 35）は、クラス階層を開いていない状態を表している。傘マークが有色であることは、そのクラス以下にインスタンスが存在することを明示的に表す。具体例としては、"Wagon" クラス 35 以下には下位クラスが存在するが、傘が閉じているため表示されていない。また、傘マーク 35 が有色であるから、この "Wagon" クラス 35 は下位クラスのいずれかにインスタンスが存在することが分かる。

【0032】

開いた傘マークで無色（白抜き）のもの（例えば参照数字 34）は、クラス階層を開いた状態を表し、かつそのクラス以下にはインスタンスが存在しないことを表す。例えば、"Sports Car" クラス 26 の一階層下に存在する "Open Car" クラス 27 が表示されている。傘マークは無色であるから、当該クラス以下には登録されているインスタンスが存在しないことが分かる。

【0033】

開いた傘マークで有色のもの（例えば参照数字 30）は、クラス階層を開いた状態を表し

10

20

30

40

50

、かつそのクラス以下にインスタンスが存在することを表す。例えば、" Vehicle " クラス 21 の一階層下に存在する " AutoBicycle " クラス 31、" Auto Mobile " クラス 23 を確認できる。その下位クラスである " Sedan " クラス 29、" Truck " クラス 25 にはインスタンスが存在するため、インスタンスそのものを表す鞠マーク 32、33 が表示されている。

#### 【0034】

上位クラスにおける下位クラスのインスタンスの有無判定は、例えば図3のフローチャートで表す処理手順に基づいて行うことができる。図3の処理手順では、各クラスの情報として、下位クラスでのインスタンス有無フラグを持たせることにしている。インスタンス有無フラグは、インスタンス新規作成の際に更新されるものとする。

#### 【0035】

図3はインスタンス有無フラグの更新処理手順を示すフローチャートである。

#### 【0036】

まず、新規にインスタンスが挿入されたクラス (class) をセットする (ステップ S1)。このクラスについて、フラグ (sub\_ins\_flg) が ON であるか否かをステップ S2 において判定する。フラグが ON と判定された場合には、下位にクラスが存在することを示して終了する。フラグが OFF であると判定された場合には、下位にクラスが存在しない。ステップ S4 においてこのフラグを ON にセットし、次にステップ S5 において上位クラスが存在するか否かを判定する。上位クラスが存在しない場合は処理を終了する。上位クラスが存在する場合はステップ S6 においてクラス (class) を上位クラスとしてステップ S2 に戻る。この処理手順によれば、インスタンスの追加があるときにだけ、最小限の範囲でフラグの更新処理が行われるため、効率的である。図2のクラス階層構造表示は、このフラグが ON であるときに、傘マークの内部が有色となる。

#### 【0037】

図4は、階層構造表示に関するより具体的な GUI 画面の構成例を示す図である。図4において、クラスの包含関係はウィンドウの領域 11 に示されている。この領域 11 内では、クラス間の包含関係を表すダイアグラムの右側端部と、上端部の表示が省略されている。

#### 【0038】

図1に示した階層構造表示装置の働きにより、階層構造表示部1とクラス情報表示部2とが連動し、階層構造表示部とプロパティリスト表示部3とが連動し、また、プロパティリスト表示部3とプロパティ情報表示部4とが連動し、表示内容が変更される。階層構造表示部1で選択されたクラスの名称、シノニム、定義などの情報をクラス情報表示部2により確認することができる。これは図4の領域12に対応する。また、階層構造表示部1で選択されたクラスで定義/継承されているプロパティのリストをプロパティリスト表示部3で確認することができる。これは、図4の領域13に対応する。このプロパティリスト表示部3で選択されたプロパティの情報は、プロパティ情報表示部4により確認することができる。プロパティの名称、シノニム、定義、データ型などの詳細情報が分かる。プロパティ情報表示部4による詳細情報の表示例を図5に示す。

#### 【0039】

「ヘリテージ (Heritage) 表示」

図6は、選択されたクラスから上位のクラスへ向かって直系の親クラス 40 だけを表示している例である。このような表示をヘリテージ (Heritage) 表示と呼ぶ。この例では、" ノート PC " クラス 41 から見た直系に相当するクラス間の親子関係が表現されている。

#### 【0040】

「検索開始ポイント (Traversal 検索) の限定」

図7は検索開始ポイントを限定するトラバーサル (Traversal) 検索を説明するための図である。階層型データベースでは、通常、階層にまたがったインスタンスの検索 (以降、Traversal 検索と呼ぶ) を、任意のクラスから行うことができる。これ

10

20

30

40

50



は、階層の上位から検索をかけることにより、どのクラスのインスタンスかをユーザが意識することなく検索が行えるという利点がある。しかしながら、検索を開始するポイントが階層の上位であればあるほど、検索対象となるクラス、およびインスタンスが爆発的に増大する。このために検索時間がかかり、システムのパフォーマンスを低下させる恐れがある。そこで図7のように検索開始ポイントを限定する。これによれば膨大なデータに対する検索を回避できるようになり、システムのパフォーマンスの低下を防ぐことができる。

#### 【0041】

検索開始ポイントは、インスタンスを持つクラスに限定することもできる。インスタンスは多くの場合、ツリーの末端近くのクラスに存在するため、検索対象クラスを少なく抑えることができる。図7の例で言うと、“Passenger Car”、“Sedan”、“Truck”の靴マークが付与された三つのクラス24、29、25がトラバーサル検索の対象となる。

#### 【0042】

また、システムアドミニストレータや辞書設計者などが、自クラスにインスタンスを含まなくとも、下位にインスタンスを含むようなクラスに対して、データベースや検索の負荷を考慮して、人的に設定することも可能である。図7の例でいうと、“Auto Bicycle”、“Sports Car”、“Open Car”など無色の傘マークまたは、末端のクラスで靴マークがついていないクラスに関しては、トラバーサル検索の対象とはならないが、他についてはその対象として、人的に検索開始ポイントを設定することが可能である。検索開始ポイントは、明示的にマークを挿入することでユーザに表示する。図7の“Auto Mobile”クラスに付与されているルーペマーク71がこれに相当する。“Auto Mobile”クラスからトラバーサル検索を行った場合、“Passenger Car”と“Sedan”、“Wagon”から以下、および“Truck”の各クラスに対して、検索条件に合ったインスタンスの検索を行うことができる。

#### 【0043】

また、図8の処理手順に従い、検索開始ポイントを自動的に設定することも可能である。図8の処理手順では、自クラスにはインスタンスを含まないが、直下の子クラスの2箇所以上にインスタンスが存在する場合に、当該クラスを検索開始ポイントとする。検索開始ポイントの設定は、例えば、クラスの情報として各クラスに持たせ、インスタンスを挿入するタイミングで行なう。2つ以上の直下の子クラスにインスタンスが存在する場合、これを検索開始ポイントとする。

#### 【0044】

class.chld\_insは、クラス(class)の直下の子クラスの、インスタンスの有無を示す変数である。初期値は、NONE(直下の子クラスにインスタンスが全くない)とする。一つの直下の子クラスにだけインスタンスが存在する場合には、該変数をONEとし(ステップS7)、2つ以上の直下の子クラスにインスタンスが存在する場合には、MULTとなる(ステップS6)。これが、MULT(複数)の場合、開始ポイントとする。図7の例では、“Auto Mobile”クラスは、直下のクラスである“Passenger Car”と“Truck”にインスタンスが存在するため、図8の処理手順に従って検索開始ポイントとして設定される。

#### 【0045】

##### 「部分継承表示」

図9は、マルチインヘリタンスを伴う階層構造の一例を示す図である。例えば、ISO13584では、単純木の階層構造のみを規定しており、マルチインヘリタンス(多重継承)は規定されていない。しかしながら、マルチインヘリタンスと類似するものとして、別のクラスの一部を継承することができるケースオブ(Case Of)なる概念が知られている。これは、別のクラスで定義されたプロパティリストの一部を継承することができるというものである。

10

20

30

40

50

## 【0046】

図9の階層構造例では、実線が通常の継承関係を表し、破線90がケースオブ(Case Of)による部分継承を表している。"Hybrid Vehicle"クラスは、"Electric Vehicle"クラスの子クラスであるが、ケースオブ(Case Of)による部分継承によって同時に"Sedan"クラスのプロパティを継承している。

## 【0047】

図10は、階層構造中の部分継承を明示的に示す表示例である。ここでは、"Hybrid Vehicle"クラス100を"Electric Vehicle"クラス101の子クラスとしてのみ表示している。この表示例では、"Sedan"クラス100の下位には子クラスは存在しないから、部分継承しているクラスが存在するのどうか分からない。しかし、"Hybrid Vehicle"クラス100が別のクラスから部分継承しているクラスであることを明示するために、他のクラスとは異なる表記(ここではクラス名称を例えば斜字体表記としている)としている。

10

## 【0048】

一方、図11は図10とは異なる部分継承の他の表示例を示したものである。この例では、"Hybrid Vehicle"クラス110を"Sedan"クラス111の子クラスのように表示している。"Sedan"クラス111よりも下位のクラスである"Hybrid Vehicle"クラス110は、"Sedan"クラス111の正規の子クラスではなく、ケースオブ(Case Of)により部分継承されている。このことを明示的に示すために、参照(Ref)マーク112を付与している。また、階層構造表示部1は、このRefマーク付のクラスが選択された場合、正規のクラスへジャンプする機構を備えており、ユーザは、"Sedan"クラス111を起点としても検索を行っていくことができる。

20

## 【0049】

## 「色別表示」

図12は、階層構造表示における色別表示を説明するための図である。階層構造表示部1により表示される階層構造の各階層の色と、プロパティリスト表示部3におけるプロパティリストの先頭マークの色とを同色で表示する。このような色別表示によれば、現在の選択クラス121からみた上位クラスのどの階層でプロパティリスト中の各プロパティが定義されているのかを、一目で確認することができ、極めて良好な視覚効果が得られる。

30

## 【0050】

## 「部分継承元クラス参照表示」

図13は、部分継承先のクラスから、部分継承したプロパティの出所クラスへの参照表示を示す図である。図13において、参照数字21は階層構造表示、22はプロパティリスト表示、23はプロパティ情報表示の例である。階層構造表示21における"Hybrid Vehicle"クラスおよびプロパティリスト表示22における"engine\_type"プロパティは、ユーザによって選択されている項目であり、ハイライト表示されている。

## 【0051】

また、プロパティリスト表示22において表示されているプロパティの左端に示すCマーク25は、このプロパティが部分継承に係るものであることを明示するためのマークである。ユーザは、この"Hybrid Vehicle"クラスが部分継承している"engine\_type"プロパティの詳細情報をプロパティ情報表示23を参照することで、このプロパティは"Sedan"クラスから継承されたものであることを確認できる。ここで、Jump(ジャンプ)ボタン24をユーザが押すと、階層構造表示21においては部分継承したプロパティの出所クラス"Sedan"クラスにフォーカスが切り替わり、これと同時にプロパティリスト表示22におけるプロパティリストが"Sedan"クラスのプロパティリストに表示更新される。

40

## 【0052】

## 「バーチャル(仮想)・ルート表示」

50

階層構造表示 21 は、通常、一つの階層構造を表示する。これは、上位クラスを持たないルート (ROOT) を源とするツリーに相当する。良く知られている C++ や Java 言語などに設けられたグラフィックライブラリなどでは、複数ツリー表示は、画面の切り替えによって行われる。このため、そのうちの一つのツリーしか一度に表示することができない。一方、本発明の実施形態では、複数のツリーに対してバーチャル・ルートを設け、それに対応するクラスを作成することにより、作成したバーチャルルートクラスの下位クラスとして、それぞれのツリーのルートを表現する。

#### 【0053】

図 14 は、階層構造表示部におけるバーチャル・ルートを設けた場合の表示例を示している。バーチャル・ルート 140 は通常のクラスと同等に扱われる。これまでに説明した本実施形態に係わるすべての機能を、バーチャル・ルート 140 下においても通常クラスと同様に利用することができる。

#### 【0054】

例えば、図 15 は、バーチャル・ルート 150 の一階層下の各ツリーのルート 151, 152 のみを初期表示するように設定したものである。

#### 【0055】

図 16 は、初期展開クラスの設定例を示す図である。ここでは、デフォルトの展開階数は「TREE\_OPEN=6」と設定されている。また、初期時に展開表示する個々のクラスは、「TREE\_OPEN\_CLASS」以下に、クラスの識別子 160 を列挙することで指定することができる。クラスの展開は、デフォルトの展開階数を原則とするが、このようにクラスの指定があれば、その枝部分については、該クラスまでの展開とする。

#### 【0056】

以上説明した本発明の実施形態によれば、階層構造そのものについての情報 (包含関係) のみならず、インスタンスの有無やプロパティリストなどを含むクラスに関する情報、プロパティに関する情報を効果的に提示するように表示部に配置する。したがって、ユーザが操作しやすいインターフェースを提供できる。たとえば、階層表示するクラス範囲を、ユーザあるいは提供者が意図的に指定して限定することもでき、複雑な階層構造を必要な部分だけに絞ってユーザに提供することができる。また、検索開始ポイントを限定してユーザに提供することで、システムに負荷がかからない検索を行わせることもできる。

#### 【0057】

なお、本発明は上述した実施形態に限定されず種々変形して実施可能である。例えば、本発明はクラス間に継承が発生するような階層型データベースのみならず、種々のシステムにおける階層構造の表示にも適用可能である。例えば、ファイルシステムにおけるディレクトリ構造ツリーの表示にも本発明は適用可能である。例えば、ディレクトリ毎に、そのインスタンスの存在有無を明示することなどである。

#### 【0058】

#### 【発明の効果】

以上説明したように、本発明によれば、階層型データベースにおけるクラス間の包含関係を適切かつ効果的に表示することのできる階層構造表示装置および方法を提供できる。

#### 【図面の簡単な説明】

【図 1】 本発明の一実施形態に係る階層構造表示システムの概略構成を示すブロック図

【図 2】 上記実施形態に係るクラス間の包含関係を示す図

【図 3】 上記実施形態に係る下位クラスにおけるインスタンスの有無判定の処理手順の一例を示すフローチャート

【図 4】 上記実施形態に係る階層構造表示に関するより具体的な GUI 画面の構成例を示す図

【図 5】 上記実施形態に係るプロパティ情報表示部による詳細情報の表示例を示す図

【図 6】 上記実施形態に係るヘリテージ (Heritage) 表示の一例を示す図

【図 7】 上記実施形態に係るトラバーサル (Traversal) 検索を説明するための図

図

10

20

30

40

50

【図 8】 上記実施形態に係る検索開始ポイントを自動的に設定する処理手順の一例を示すフローチャート

【図 9】 上記実施形態に係るマルチインヘリタンスを伴う階層構造の一例を示す図

【図 10】 上記実施形態に係る階層構造中の部分継承の表示例を示す図

【図 11】 上記実施形態に係る部分継承の他の表示例を示す図

【図 12】 上記実施形態に係る色別表示を説明するための図

【図 13】 上記実施形態に係る部分継承先のクラスから、部分継承したプロパティの出所クラスへの参照表示を示す図

【図 14】 上記実施形態に係る階層構造表示部におけるバーチャル・ルートを設けた場合の表示例を示す図

【図 15】 上記実施形態に係るバーチャル・ルート設定時の初期表示例を示す図

【図 16】 上記実施形態に係る初期展開クラスの設定例を示す図

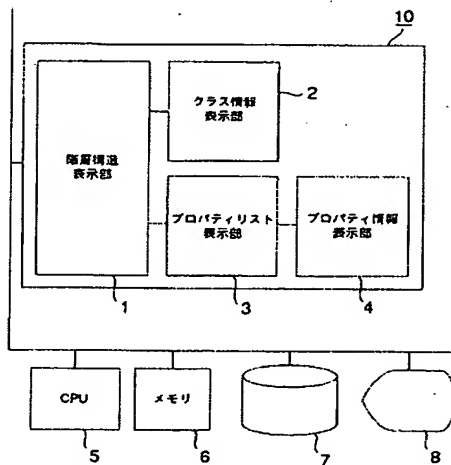
【符号の説明】

- 1…階層構造表示部
- 2…クラス情報表示部
- 3…プロパティリスト表示部
- 4…プロパティ情報表示部
- 5…CPU
- 6…メモリ
- 7…二次記憶装置
- 8…ディスプレイ

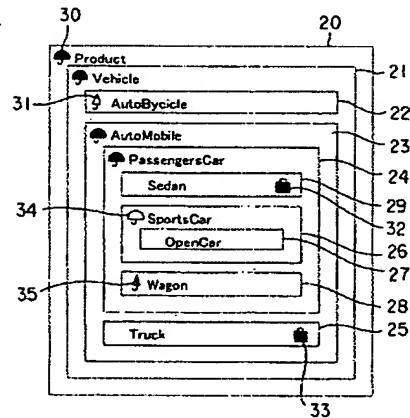
10

20

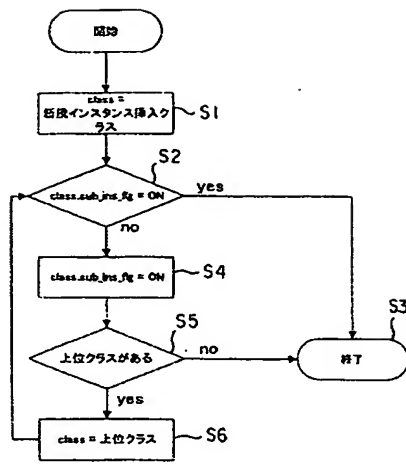
【図 1】



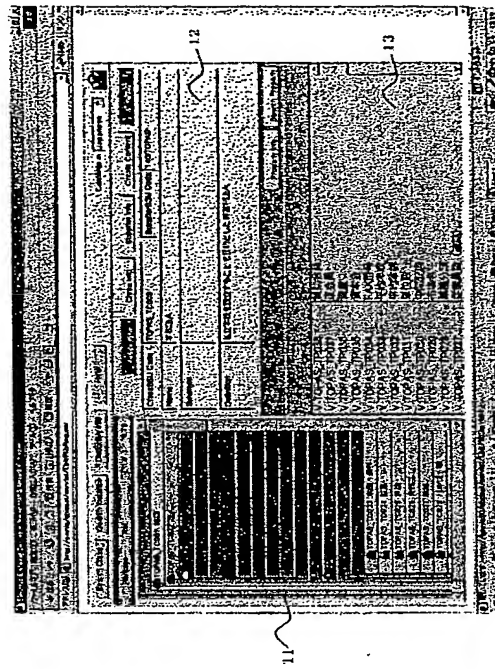
【図 2】



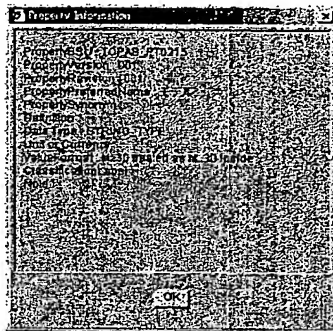
【図 3】



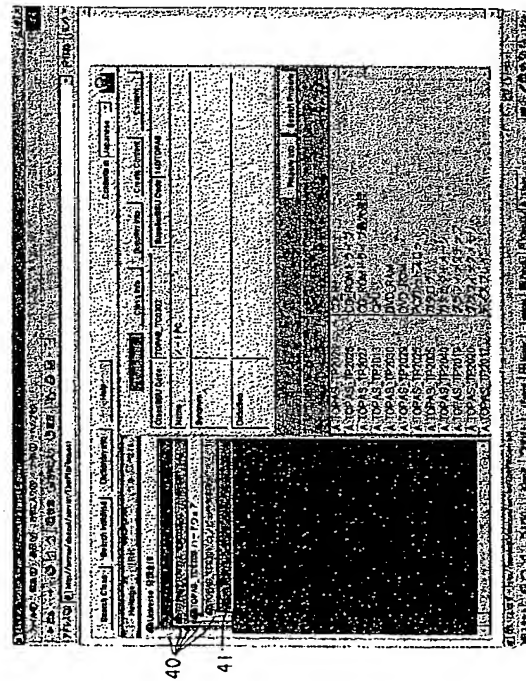
【図 4】



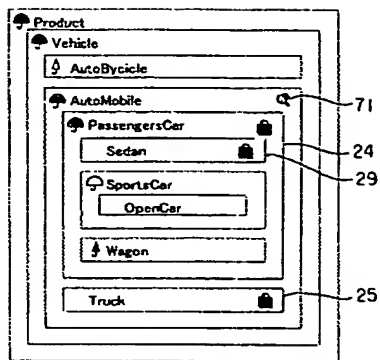
【図 5】



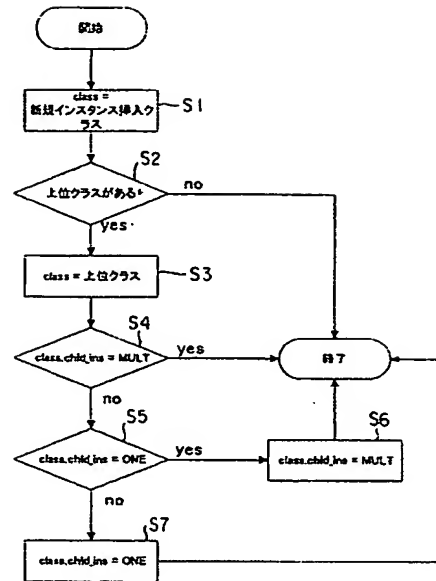
【図 6】



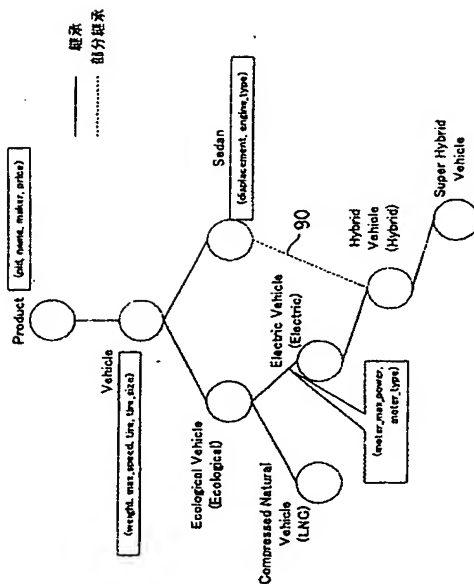
【図 7】



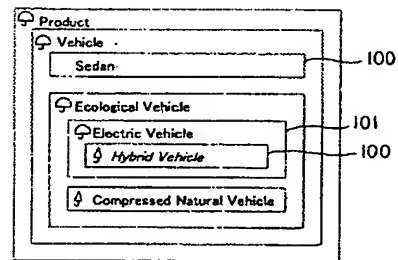
【図 8】



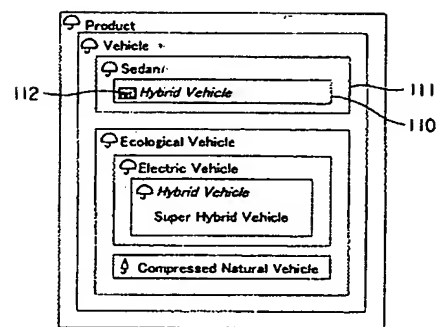
【図 9】



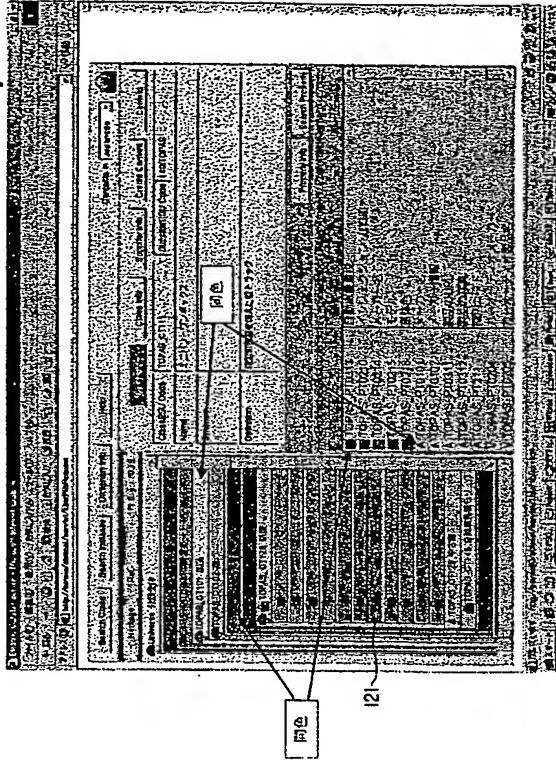
【図 10】



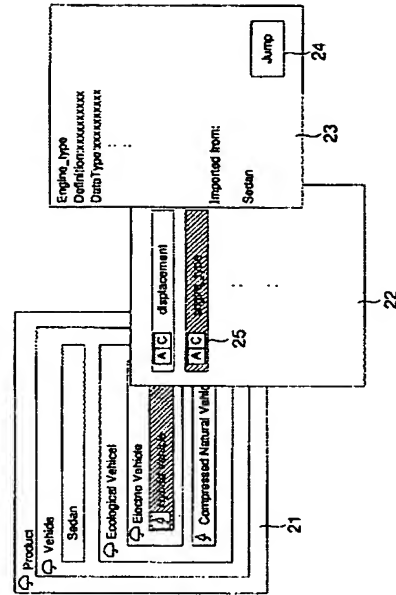
【図 11】



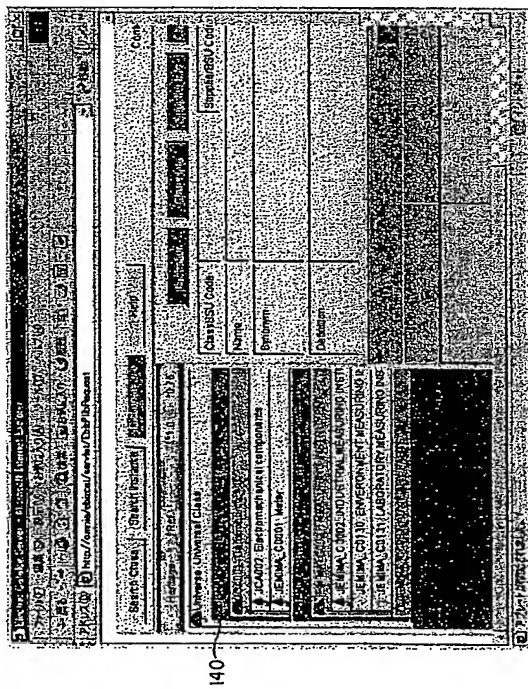
【図 1 2】



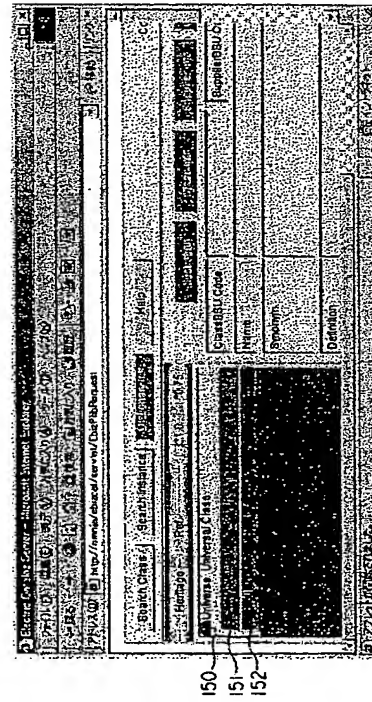
【図 1 3】



【図 1 4】



【図 1 5】



【図 16】

```
TREE_OPEN=6
TREE_OPEN_CLASS
  140/TOPAS.TOPAS_CT101
  140/TOPAS.TOPAS_CT103
  140/TOPAS.TOPAS_CT104
END
```

160



## フロントページの続き

- (74) 代理人 100070437  
弁理士 河井 将次
- (72) 発明者 南野 典子  
神奈川県川崎市幸区小向東芝町 1 番地 株式会社東芝研究開発センター内
- (72) 発明者 溝口 祐美子  
神奈川県川崎市幸区小向東芝町 1 番地 株式会社東芝研究開発センター内
- (72) 発明者 村山 廣  
神奈川県川崎市幸区小向東芝町 1 番地 株式会社東芝研究開発センター内
- Fターム(参考) 5B082 GC02

